

# Echtzeitsysteme

## Lehrveranstaltungskonzept

13. Oktober 2008

# Überblick

## Lehrveranstaltungskonzept

Einordnung

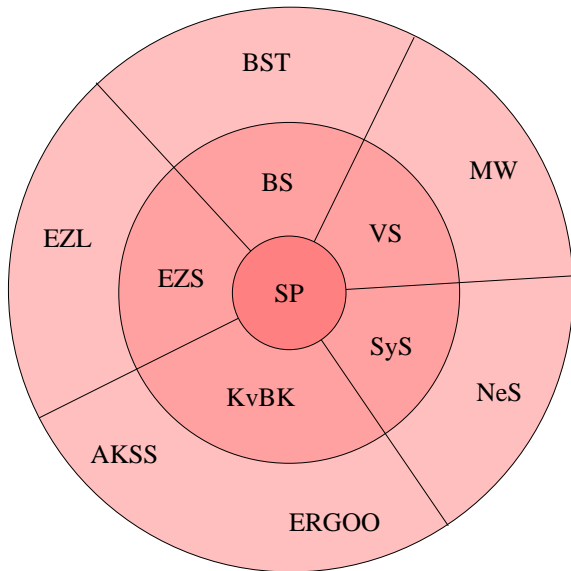
Studiengänge

Lernziele und Lehrinhalte

Voraussetzungen

Ablauf der Lehrveranstaltung

Leistungsnachweise



# Lehre@I4: *post* EZS — Aufbau und Spezialisierung

EZL Echtzeitsystemlabor

IV

BS Betriebssysteme

V/Ü

VS Verteilte Systeme

V/Ü

SyS Systemsicherheit

V/Ü

BST Betriebssystemtechnik

V/Ü

MW Middleware

V/Ü

NeS Netzwerksicherheit  $\mapsto$  I7

V/Ü

AKSS Ausgewählte Kapitel der Systemsoftware

HS

ERGOO Erlangen Research Group on

Distributed Objects and Operating Systems

KO

# Integrierte Lehrveranstaltung

$$\text{Termine } \left\{ \begin{array}{ll} \text{Vorlesung} & 1 \\ \text{Übung} & 1 \\ \text{Rechner} & 1 \end{array} \right\} \equiv 3 \times 1,5 = 4,5 \text{ Zeitstunden wöchentlich}$$

## Vor-/Nacharbeit

- ▶  $N$  Stunden wöchentlich:  $0 \leq N \leq (163,5 - X)$
- ▶  $X \ll 163,5$  ist das Zeitstundenäquivalent anderer „Pflichten“

# Fakultative Vertiefung in Informatik

## Diplom

- ▶ Informatik, IuK
- ▶ Mechatronik, Maschinenbau
- ▶ Technomathematik, Wirtschaftsinformatik

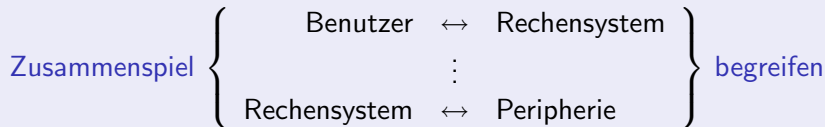
## Bachelor

- ▶ Computational Engineering (CE)

*Echtzeitsysteme findet man nicht nur in den mehr technisch ausgelegten Bereichen. Entscheidungsprozesse der Wirtschaft sind verschiedentlich ebenfalls echtzeitabhängig.*

# Lernziele

Rechensystem und seine Umgebung als **Ganzes** verstehen:



Objektorientierte **Echtzeitprogrammierung** (in Grundzügen) erleben

- ▶ betriebssystemnah praktizieren

Grundlagen von echtzeitfähigen Softwaresystemen erlernen

- ▶ fähig sein, den Grad an Echtzeitfähigkeit eines Systems zu erkennen
- ▶ seitens gegebener Anwendungen und Hard-/Softwareplattformen

# Lehrinhalte

**Vorlesung:** Vorstellung und detaillierte Behandlung des Lehrstoffs

- ▶ Grundlagen von Echtzeitsystemen
  - ▶ zeit- und ereignisgesteuerte Systeme
  - ▶ periodische und sporadische Aufgaben (engl. *tasks*)
  - ▶ Einplanung und Koordination
- ▶ Echtzeitbetriebssysteme

**Übung:** Vertiefung, Besprechung der Übungsaufgaben, Tafelübungen

- ▶ Systemprogrammierung in C/C++
- ▶ echtzeitfähige Systemprogramme

# Erforderliche Grundkenntnisse

## Grundlagen der Systemprogrammierung: [Softwaresysteme](#)

- ▶ systemnahe Programmierung in C
  - ▶ asynchrone Programmunterbrechungen, Aktivitätsträger, Prozesse
  - ▶ koordinierte Ausführung nebenläufiger Programme
- ▶ Grundzüge von Betriebssystemen

## Grundlagen der Rechnerorganisation: [Technische Informatik](#)

- ▶ maschinennahe Programmierung
  - ▶ Unterbrechungssteuerung (Pegel kontra Flanke)
  - ▶ Assemblerprogrammierung
- ▶ CPU, DMA, IRQ, MCU, NMI, PIC

## Betriebssystemkenntnisse

- ▶ schaden nie — sind fördernd, sicher hilfreich und daher erwünscht

# Vorlesungsbetrieb und Lehrmaterialien

## Vorlesungstermine:

- ▶ Montag, 14:15–15:45, H4

## Handzettel (engl. *handout*) sind verfügbar wie folgt:

- ▶ [www4.informatik.uni-erlangen.de/Lehre/WS08/V\\_EZS](http://www4.informatik.uni-erlangen.de/Lehre/WS08/V_EZS)
- ▶ **kein Skript**, bloß die Folien zum Vorlesungs- und Übungsstoff

## Fachbegriffe der Informatik (Deutsch ↔ Englisch)

- ▶ [www.babylonia.ork.uk](http://www.babylonia.ork.uk)
- ▶ [www.aktionlebendigesdeutsch.de](http://www.aktionlebendigesdeutsch.de)

# Ergänzende Literatur

-  Hermann Kopetz.  
*Real-Time Systems: Design Principles for Distributed Embedded Applications.* Kluwer Academic Publishers, 1997.
-  Jane W. S. Liu.  
*Real-Time Systems.* Prentice-Hall, Inc., 2000.
-  Wolfgang Schröder-Preikschat.  
Softwaresysteme 1. Lecture Notes.  
[www4.informatik.uni-erlangen.de/Lehre/SS07/V\\_S0S1](http://www4.informatik.uni-erlangen.de/Lehre/SS07/V_S0S1), 2007.
-  Wolfgang Schröder-Preikschat.  
Echtzeitbetriebssysteme. In Peter Liggesmeyer and Dieter Rombach, editors, *Software Engineering für Eingebettete Systeme*, chapter 15, pages 343–376. Elsevier, Spektrum Akademischer Verlag, 2005.

# Übungsbetrieb

Übungstermine Anmeldung ab sofort, Beginn ab 20.10.:

Tafel Anzahl je nach Teilnehmeranzahl

Montag	10:15–11:45	0.031
Dienstag	14:15–15:45	0.031
Donnerstag	12:15–13:45	00.156

Rechner zusätzlich zur „Heimarbeit“

Donnerstag	10:15–11:45	02.151a
------------	-------------	---------

Kontinuität und aktive Mitarbeit ist der Schlüssel zum Erfolg

- ▶ Trockenübung ohne praktische Umsetzung ist wenig ergiebig
- ▶ praktische Umsetzung ohne Trockenübung ist zu beschwerlich

# Bedeutung von Tafel- und Rechnerübungen

## Tafelübungen „*learning by exploring*“

- ▶ Besprechung der Übungsaufgaben, Skizzierung von Lösungswegen
- ▶ Vertiefung des Vorlesungsstoffes, Klärung offener Fragen

## Rechnerübungen „*learning by doing*“

- ▶ selbständiges Bearbeiten der Übungsaufgaben am Rechner
- ▶ Hilfestellung beim Umgang mit den Entwicklungswerkzeugen
- ▶ der Rechner ist **kein Tafelersatz**, die Betreuung verläuft eher passiv

*Der, die, das.  
Wer, wie, was?  
Wieso, weshalb, warum?  
Wer nicht fragt, bleibt dumm!*



# Leistungskontrolle

**unbenoteter Schein:** obligatorisch für Rücksprache zum benoteten Schein

- ▶ erfolgreiche Bearbeitung aller Übungsaufgaben

**Prüfung:** mündlich

- ▶ Themenschwerpunkte je nach Teilnahme am Übungsbetrieb
  - vorlesungslastig** bei bestandenem Schein
  - übungslastig** sonst
- ▶ den unbenoteten Schein zu machen, wird dringend empfohlen

# Übungsscheinkriterien

**Abgabefristen** von Übungsaufgaben sind **strikt** einzuhalten

- ▶ elektronisch gestützter, semi-automatischer Abgabevorgang:
  - ▶ prüft die Aufgabenlösungen auf Plausibilität
  - ▶ testet auf Abschreiben
- ▶ stichprobenartig werden einzelne Aufgaben genauer durchkorrigiert

**Scheinvergabe** nur bei ausreichender Bearbeitung der Übungsaufgaben

- ▶ Plausibilitätsprüfung und Abschreibetest  $\leadsto$  OK
- ▶ die Stichprobenkorrektur muss ein positives Ergebnis geliefert haben
- ▶ ggf. entscheidet eine persönliche Rücksprache über Erfolg/Misserfolg

# Rücksprache und Prüfung

**Rücksprache** für einen benoteten Übungsschein, 30 Minuten

- ▶ Thema ist Stoff der Übung aber auch der Vorlesung
- ▶ Einzelgespräch, Termin direkt mit dem Dozenten vereinbaren

**Prüfung:** mündlich, 15–30 Minuten

- ▶ Thema ist Stoff der Vorlesung aber auch der Übung
- ▶ Einzelprüfung, Termin direkt mit dem Dozenten vereinbaren

Präsenz und **aktive Mitarbeit** machen Rücksprache bzw. Prüfung „leicht“

- ▶ Programme im Team entwickeln, aber selbst zum Laufen bringen

www4.informatik.uni-erlangen.de/\*

## Dozent

- ▶ Wolfgang Schröder-Preikschat (~wosch)

## Mitarbeiter

- ▶ Niko Böhm (~boehm)
- ▶ Fabian Scheler (~scheler)
- ▶ Peter Ulbrich (~ulbrich)

# Fragen...

42